Remarks

Applicant respectfully requests that this Response After Final Action be admitted under 37 C.F.R. § 1.116.

Applicant submits that this Response presents claims in better form for consideration on appeal. Furthermore, applicant believes that consideration of this Response could lead to favorable action that would remove one or more issues for appeal.

No claims have been amended. No claims have been canceled. Therefore, claims 1-3, 5-15 and 18, 19 and 21-28 are now presented for examination.

Claims 1, 7, 23, and 26 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Tremaine (U.S. Patent No. 6,775,751) in view of Dye et al. (U.S. Patent No. 6,879,266) and Goldberg (U.S. Patent No. 7,035,656). Applicant submits that the present claims are patentable over Tremaine in view of Dye and Goldberg.

Tremaine discloses a method and structure for reducing access latency and contention in a processing. The method detects when the amount of available memory is outside a prescribed range, and responsively selects data blocks for compression (to add to the available memory,) or decompression (to use surplus available memory for uncompressed data,) until the amount of available memory is within the prescribed range. When data blocks are compressed, a DOC is determined and stored as an attribute in the directory entry associated with the data block. A most recently used list of recent data block addresses prevents those, as well those data blocks with poor DOC attributes, from being selected for recompression. All zero data blocks are detected to avoid standard compression/decompression overhead. See Tremaine at Abstract.

Dye discloses compressing data with a fixed compression ratio. See Dye at Col. 33, ll. 55-62.

Goldberg discloses a network controller for controlling a message over a communication network constantly monitors actually transmitted message from or to a particular user. If the controller determines a particular phrase is transmitted more than a predetermined number of times, the controller updates support data and transmits the data in a compressed format with support data. The network controller may send an updated support data in response to a request from user equipment. See Goldberg at Abstract.

Claim 1 of the present application recites:

> A method comprising:
> receiving a string of data symbols; and
> compressing the string of data into a fixed sized compressed data block having a plurality of compressed symbols and dictionary elements, <u>the compressed symbols and dictionary elements having a fixed length and a fixed offset within the compressed data block, wherein the number of dictionary elements is automatically derived from a number of leading bits in the string of data</u>.

Applicant submits that Tremaine, Dye and Goldberg all fail to disclose or suggest compressed symbols and dictionary elements having a fixed length and a fixed offset within a compressed data block. The Examiner asserts that Dye discloses such a feature at col. 7, ll. 20-30 and col. 33, ll. 56-62. The passages referred to by the Examiner disclose:

> The MemoryF/X Technology preferably includes novel parallel compression and decompression engines designed to process stream data at more than a single byte or symbol (character) at one time. These parallel compression and decompression engines modify the single stream dictionary based (or history table based) data compression method described by Lempel and Ziv

> to provide a scalable, high bandwidth compression and decompression operation. The parallel compression method examines a plurality of symbols in parallel, thus providing greatly increased compression performance.

Dye at col. 7, ll. 20-30.

> Alternatively, some applications such as consumer appliances and embedded DRAM require a "fixed" compression ratio in order to accommodate a fixed size memory environment. Fixed compression ratio allows the software to allocate memory in a known size and also compensates for overflow of data past the physical limit of the memory size. In this alternate embodiment, where a fixed compression ratio is required, the lossy algorithm is easily changed to eliminate special cases, which in the preferred embodiment allow a better compression ratio.

Dye at col. 33, ll. 56-62.

The above-passages discloses a "fixed" compression ratio that accommodates a fixed size memory environment. Nevertheless, there is no disclosure or suggestion of such a "fixed" compression ratio including ***compressed symbols*** *and* ***dictionary elements having a fixed length and a fixed offset <u>within</u> a compressed data block.***

Moreover, applicant submits that Tremaine, Dye and Goldberg all fail to disclose or suggest a number of dictionary elements automatically derived from a number of leading bits in a string of data. The Examiner asserts that Dye discloses such a feature at col. 24, ll. 2-65. The passage referred to by the Examiner discloses:

> FIG. 6A--Prior Art
>
> Prior art has made use of the LZ compression algorithm for design of computer hardware, but the bandwidth of the data stream has been limited due to the need to serially review the incoming data to properly generate the compressed output stream. FIG. 6A depicts the prior art normal history table implementation.

The LZ compression algorithm attempts to reduce the number of bits required to store data by searching that data for repeated symbols or groups of symbols. A hardware implementation of an LZ77 algorithm would make use of a history table to remember the last n symbols of a data stream so that they could be compared with the incoming data. When a match is found between the incoming stream and the history table, the matching symbols from the stream are replaced by a compressed symbol, which describes how to recover the symbols from the history table.

FIG. 6B--Parallel Algorithm

The preferred embodiment of the present invention provides a parallel implementation of dictionary based (or history table based) compression/decompression. By designing a parallel history table, and the associated compare logic, the bandwidth of the compression algorithm can be increased many times. This specification describes the implementation of a 4 symbol parallel algorithm which results in a 4 times improvement in the bandwidth of the implementation with no reduction in the compression ratio of the data. In alternate embodiments, the number of symbols and parallel history table can be increased and scaled beyond four for improved parallel operation and bandwidth, or reduced to ease the hardware circuit requirements. In general, the parallel compression algorithm can be a 2 symbol parallel algorithm or greater, and is preferably a multiple of 2, e.g., 2, 4, 8, 16, 32, etc. The parallel algorithm is described below with reference to a 4 symbol parallel algorithm for illustrative purposes.

The parallel algorithm comprises paralleling three parts of the serial algorithm: the history table (or history window), analysis of symbols and compressed stream selection, and the output generation. In the preferred embodiment the data-flow through the history table becomes a 4 symbol parallel flow instead of a single symbol history table. Also, 4 symbols are analyzed in parallel, and multiple compressed outputs may also be provided in parallel. Other alternate embodiments may contain a plurality of compression windows for decompression of multiple streams, allowing a context switch between decompression of individual data blocks. Such alternate embodiments may increase the cost and gate counts with the advantage of suspending

> current block decompression in favor of other block decompression to reduce latency during fetch operations. For ease of discussion, this disclosure will assume a symbol to be a byte of data. Symbols can be any reasonable size as required by the implementation. FIG. 6B shows the data-flow for the parallel history table.
>
> FIG. 7--High Level Flowchart of the Parallel Compression Algorithm
>
> FIG. 7 is a high-level flowchart diagram illustrating operation of the parallel compression algorithm in the preferred embodiment. Steps in the flowchart may occur concurrently or in different orders.
>
> In step 402, the method maintains a history table (also called a history window) comprising entries, wherein each entry may comprise one symbol. The history table is preferably a sliding window that stores the last n symbols of the data stream.
>
> In step 404 the method maintains a current count of prior matches which occurred when previous symbols were compared with entries in the history table. A count is maintained for each entry in the history table.

Dye at col. 24, ll. 2-65.

The above-passage discloses a parallel implementation of dictionary based compression/decompression using a 4 symbol parallel algorithm, resulting in a 4 times improvement in the bandwidth of the implementation with no reduction in the compression ratio of the data. Also, it is disclosed that in alternate embodiments the number of symbols and parallel history table can be increased and scaled beyond four for improved parallel operation and bandwidth, or reduced to ease the hardware circuit requirements. However, nowhere in the above-passage relied on by the Examiner is there a disclosure, or reasonable suggestion of *a number of dictionary elements* ***<u>automatically derived</u>*** ***from a number of leading bits in a string of data***.

Because Tremaine, Dye and Goldberg each fail to disclose or suggest compressed symbols and dictionary elements having a fixed length and a fixed offset within a

compressed data block, <u>or</u> a number of dictionary elements in a compression block being automatically derived from a number of leading bits in a string of data, any combination of Tremaine, Dye and Goldberg would not disclose or suggest such features. Accordingly, claim 1, and its dependent claims, is patentable over a combination of Tremaine, Dye and Goldberg.

Independent claims 7, 23 and 26 include limitations similar to those recited in claim 1. Thus, claims 7, 23 and 26, and their respective dependent claims, are patentable over a combination of Tremaine, Dye and Goldberg.

Claims 2-3, 5-6, 8-9, 11-12, 24-25, and 27 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Tremaine in view of Dye et al., and Goldberg as applied to claims 1, 7, 23, and 26 above, and further in view of Castelli et al. (U.S. Patent No. 6,847,315). Applicant submits that the present claims are patentable over Tremaine, Dye and Goldberg even in view of Castelli.

Castelli discloses a method and structure that stores and/or transmits and receives data in compressed form. Retrieval latencies are reduced by selectively transmitting a portion of the data in uncompressed form. When the apparatus is part of a computer architecture supporting main memory compression, a selected L2 cache line belonging to the unit of main memory compression is kept uncompressed. To minimize decompression latency, the uncompressed L2 cache line is stored with the compressed-memory directory. Alternatively, the uncompressed L2 cache line is stored in the compressed memory together with the rest of the memory compression unit it belongs to. See Castelli at Abstract.

Nevertheless, Castelli does not disclose or suggest compressed symbols and dictionary elements having a fixed length and a fixed offset within a compressed data block, or a number of dictionary elements in a compression block being automatically derived from a number of leading bits in a string of data. As discussed above, Tremaine, Dye and Goldberg do not disclose or suggest such features. Therefore, any combination of Tremaine, Dye, Goldberg and Castelli would not disclose or suggest the features. Thus, the present claims are patentable over a combination of Tremaine, Dye, Goldberg and Castelli.

Claims 13-14 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Tremaine in view of Dye et al., and Goldberg and further in view of Castelli as applied to claims 8 and 24 above, and further in view of Franaszek et al. (U.S. Patent No. 5,729,228). Applicant submits that the present claims are patentable over a combination of Tremaine, Dye, Goldberg, Castelli and Franaszek because each reference fails to disclose or suggest compressed symbols and dictionary elements having a fixed length and a fixed offset within a compressed data block, or a number of dictionary elements in a compression block being automatically derived from a number of leading bits in a string of data.

Claims 15, 19-21 and 28 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Dye in view of Goldberg. Applicant submits that the present claims are patentable over Dye in view of Goldberg.

Claim 15 of the present application recites

> A method comprising:
> receiving a fixed offset compressed data block having a plurality of dictionary elements and compressed symbols; and

decompressing each of the compressed symbols in parallel, by:
analyzing encoded tag bits within a compressed symbol; and
decompressing the compressed symbol to form a symbol based upon a type of compression indicated by the encoded tag bits, wherein each of the compressed symbols are decompressed simultaneously.

Similar to the reasons discussed above with respect to claim , Applicant submits that neither Dye nor Goldberg disclose or suggest ***a fixed offset*** *compressed data block.* Thus, the combination of Dye and Goldberg doe not disclose or suggest all of the limitations of claim 15, or its dependent claims.

Independent claim 19 includes limitations similar to those recited in claim 15. Thus, claim 19, and its dependent claims, are also patentable over Dye in view of Goldberg.

Claims 18 and 22 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Dye in view of Goldberg as applied to claims 15 and 19 above, and further in view of Castelli. Applicant submits that any combination of Dye, Goldberg and Castelli would fail to disclose or suggest a fixed offset compressed data block. Therefore, the present claims are patentable over the combination of Dye, Goldberg and Castelli

Applicant respectfully submits that the rejections have been overcome, and that the claims are in condition for allowance. Accordingly, applicant respectfully requests the rejections be withdrawn and the claims be allowed.

The Examiner is requested to call the undersigned at (303) 740-1980 if there remains any issue with allowance of the case.

Please charge any shortage to our Deposit Account No. 02-2666.

Respectfully submitted,
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: May 19, 2008

Mark L. Watson
Reg. No. 46,322

1279 Oakmead Parkway
Sunnyvale, California 94085-4040
(303) 740-1980